CLAIM LISTING

This listing of claims will replace all prior versions, and listings of claims in the application:

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method of reducing a size of a netlist for a target architecture for simulation, comprising:

create a netlist of objects specifying each object for the target hardware architecture, wherein the target hardware architecture is a field programmable gate array;

identify objects specific to the target hardware architecture that are repeated to identify potential dummy objects;

create a list of objects, from the netlist of objects, that are used by a circuit design to be implemented in the target hardware architecture;

form a list of unused objects in the target hardware architecture from the netlist of objects and the list of objects used by the circuit design;

replace at least one object in the netlist of objects for the target hardware architecture that is also specified in the list of unused objects and which is identified as a potential dummy object with an appropriate dummy object to form a modified netlist by removing functional hardware description language from the object; and

simulate the modified netlist by simulating each object of the modified netlist inclusive of each dummy object, wherein for each dummy object, a signal provided to the dummy object is fed through the dummy object unchanged.

2. (Previously Presented) The method of claim 1, wherein the step of forming the list of unused objects comprises the steps of subtracting the list of objects used by the circuit design from the netlist of objects.

3. (Previously Presented) The method of claim 1, wherein the step of replacing at least one object comprises the step of replacing objects in the netlist of objects for the target hardware architecture that are also specified in the list of unused objects that are repeated with the appropriate dummy objects to form the modified netlist.

4. (Previously Presented) The method of claim 1, wherein the step of replacing at least one object comprises the step of replacing each object in the netlist of objects for the target hardware architecture that is also specified in the list of unused objects with the appropriate dummy object to form the modified netlist.

5. (Previously Presented) The method of claim 1, wherein the step of forming the list of unused objects comprises the steps of parsing a file to extract a list containing object names for all used instances for the target hardware architecture and parsing the netlist for the target architecture.

6. (Previously Presented) The method of claim 1, wherein the step of replacing comprises replacing a type of an instance for an object found in the repeated list of objects with a type for a corresponding dummy object if the object found in the repeated list is not on the list containing object names for used objects.

7. (Previously Presented) The method of claim 1, wherein the method further comprises the step of parsing the netlist for the target hardware architecture line by line and forming a modified netlist with the appropriate dummy objects when all lines of the netlist have been parsed.

8. (Original) The method of claim 7, wherein the method further comprises the step of feeding through a signal unchanged when simulating the appropriate dummy object during a simulation process using the modified netlist.

9. (Previously Presented) The method of claim 1, wherein the step of forming the list of unused objects comprises the step of manually composing a list of repeated root objects specific to the target hardware architecture.

10. (Original) The method of claim 1, wherein the step of replacing comprises the step of emptying a hardware description language version of a repeated object to form an object devoid of an explicit functional mapping of an input to an output.

11. (Previously Presented) A method of processing signals with a modified netlist within a software-based logic simulation tool comprising the steps of:

creating a list of repeated objects specific to a target hardware architecture that are repeated objects;

emptying the repeated objects found on the list of repeated objects forming a plurality of dummy objects;

parsing a programmatic circuit description to be implemented in the target hardware architecture to extract a list containing object names for all used objects for the target architecture;

parsing a netlist of objects line by line for the target architecture;

replacing any object in the netlist with a corresponding dummy object from the plurality of dummy objects if the object in the netlist is not on the list containing object names for all used objects to form a modified netlist; and

simulating the modified netlist.

12. (Original) The method of claim 11, wherein the method further comprises the step of feeding through a signal unchanged when simulating the plurality of dummy objects during a simulation process using the modified netlist.

13. (Original) The method of claim 11, wherein the step of emptying the repeated objects comprises the step of emptying a hardware description language version of the repeated objects to form a plurality objects devoid of logic.

14. (Previously Presented) The method of claim 11, wherein the step of parsing the file comprises the step of parsing a file containing hierarchical path names to memory blocks of a field programmable gate array forming the target hardware architecture.

15. (Original) The method of claim 14, wherein the method further comprises the step of generating the file containing the hierarchical path names by converting bitstream names into Verilog hierarchical path names.

16. (Currently Amended) A machine readable storage, having stored thereon a computer program having a plurality of code sections executable by a machine for causing the machine to perform the steps of:

create a netlist of objects specifying each object for a target hardware architecture, wherein the target hardware architecture is a field programmable gate array;

identify objects specific to the target hardware architecture that are repeated to identify potential dummy objects;

create a list of objects used by a circuit design to be implemented in the target hardware architecture;

form a list of unused objects in the target hardware architecture from the netlist of objects and the list of objects used by the circuit design;

replace at least one object in the netlist of the target hardware architecture that is also specified in the list of unused objects and which is identified as a potential dummy object with an appropriate dummy object to form a modified netlist by removing functional hardware description language from the object; and

simulate the modified netlist by simulating each object of the modified netlist inclusive of each dummy object, wherein for each dummy object, a signal provided to the dummy object is fed through the dummy object unchanged.

17. (Previously Presented) The machine readable storage of claim 16, wherein the computer program further has a plurality of code sections executable by the machine for causing the machine to perform the step of parsing the netlist for the target hardware architecture line by line and forming a modified netlist with one or more of the appropriate dummy objects when all lines of the netlist have been parsed.

18. (Currently Amended) A system for simulating a circuit design for target hardware architectures for implementation on field programmable gate arrays, comprising:

means for creating a netlist of objects specifying each object for the target hardware architecture, wherein the target hardware architecture is a field programmable gate array;

means for identifying objects specific to the target hardware architecture that are repeated to identify potential dummy objects;

means for creating a list of objects used by the circuit design in the target hardware architecture;

means for forming a list of unused objects in the target hardware architecture from the netlist of objects and the list of objects used by the circuit design;

means for replacing at least one object in the netlist of the target hardware architecture that is also specified in the list of unused objects and which is identified as a potential dummy object with an appropriate dummy object to form a modified netlist by removing functional hardware description language from the object; and

means for simulating the modified netlist by simulating each object of the modified netlist inclusive of each dummy object, wherein for each dummy object, a signal provided to the dummy object is fed through the dummy object unchanged.

19. (Previously Presented) The system of claim 18, further comprising means for parsing the netlist for the target hardware architecture line by line and forming a modified netlist with a plurality of dummy objects when all lines of the netlist have been parsed.

20. (Original) The system of claim 18, wherein the system uses a Verilog version of a hardware description language.